REMARKS

Claims 1-24 remain pending in this application.

The Examiner has rejected Claims 1-3, 5, 7, 11, 12, 14-17, 22-24 under 35 U.S.C. 112, second paragraph.

Claims 1-3, 5, 7, 12, 14-17, 22-24 have been amended to clarify the subject matter regarded as the invention.

Regarding Claim 11, the antecedent basis for "the cross validation criteria" is found in Claim 1.

The Examiner has rejected Claims 1-11, 13-24 under 35 U.S.C. 102(b) as being anticipated by Ratsaby et al (referred to as Ratsaby).

The rejection is respectfully traversed. With respect to Claim 1 and 24, Ratsaby teaches methods for selecting robust model by computing upper and lower bounds to the error of approximation by neural networks (page 57, col 2, lines 9-13), but does not mention selecting an associated set of weights of the modeling function with the value of a complexity parameter that best satisfies a cross validation criteria. Although Ratsaby teaches that approximation error may be reduced by simply increasing the complexity of the model (page 58, col 2, lines 17-19), he does not describe how to select a set of weights of a robust modeling function with the value of a complexity parameter that best satisfies a cross validation criteria. Ratsaby mentions the usage of a "complexity index" (page 59, col 2, 16). However, Ratsaby does not suggest how such an index would be used to select a robust model. There is no teaching that the complexity index should satisfy a cross validation criteria as recited in Claims 1 and 24.

Claim 1 and 24 are therefore believed to be allowable.

Claims 2-22 depend from Claim 1 and are believed to be allowable for the same reasons described above.

a

Specifically regarding Claim 21, Ratsaby does not disclose that the cross validation criteria could be maximizing lift. Lift is described in the specification as comparing the performance of the selected group compared to the expected performance of a randomly selected group and a group selected by a "wizard" having perfect knowledge ("Robust Modeling", U.S. Patent Application No. 09/418,537, Attorney Docket No. KXENP001, page 22, line 3-5). Because there is no suggestion of lift being used as a cross validation criteria, Claim 21 is believed to be allowable over Ratsaby.

Regarding Claim 23, Ratsaby teaches methods for selecting robust model by computing upper and lower bounds to the error of approximation by neural networks (page 57, col 2, lines 9-13), but does not teach determining an optimal value of the complexity parameter that minimizes the cross validation error. Although Ratsaby teaches that approximation error may be reduced by simply increasing the complexity of the model (page 58, col 2, lines 17-19), he does not describe how to select a robust modeling function by determining an optimal complexity parameter that minimizes the cross validation error. Ratsaby mentions the usage of a "complexity index" (page 59, col 2, 16). However, Ratsaby does not suggest how such an index would be used to select a robust model. There is no teaching that the complexity index should minimize a cross validation error as recited in Claim 23.

Claim 23 is therefore believed to be allowable.

The Examiner has rejected Claim 12 under 35 U.S.C. 102(b) as being anticipated by Brent. Brent teaches how a neural net may be derived from a decision tree and how to construct a suitable decision tree (page 1, col 1, line 36), but does not teach selecting an associated set of weights of the modeling function with the value of a complexity parameter that best satisfies a cross validation criteria. Claim 12 is therefore believed to be allowable.

a

Attached hereto is a marked-up version of the changes made to the specification and claims by the current amendment. The attached page is captioned "Version with markings to show changes made."

Reconsideration of the application and allowance of all Claims are respectfully requested based on the preceding remarks. If at any time the Examiner believes that an interview would be helpful, please contact the undersigned.

Respectfully submitted,

Lee Van Pelt
Registration No. 38,352
V 650 903-3503
F 650 903-3501

VAN PELT AND YI, LLP
4906 El Camino Real, Suite 205
Los Altos, CA 94022

a

VERSION WITH MARKINGS TO SHOW CHANGES MADE

AMENDMENTS TO THE SPECIFICATION

On page 19, starting on line 1:

The remainder of the data set shown below is then used to select an optimum Lambda. The output of the model is compared to the output specified in the [training] validation set for each input. A goal is defined for the comparison and a search is performed by varying Lambda to determine an optimum value of Lambda that best achieves the goal. Selecting an optimum Lambda is described in further detail below.

On page 19, starting on line 16:

The weights [(i] wi shown above may be derived by minimizing the regularization function for a given Lambda. Next, a technique will be described for determining an optimum value of Lambda.

On page 22, starting on line 11:

In some embodiments, once an optimal Lambda is selected, a final model is retrained using the optimum [Lamba] Lambda. The retraining is done by using both the training data set and the cross validation data set to determine the final set of weights ω.

a

AMENDMENTS TO THE CLAIMS

1. (Amended) A method of generating a robust model of a system comprising:

selecting a modeling function having a set of weights wherein the modeling function has a complexity that is determined by a complexity parameter;

for each of a plurality of values of the complexity parameter, determining an associated set of weights of the modeling function such that a training error is minimized for a training data set;

determining an error for a cross validation data set for each set of weights associated with one of the plurality of values of the complexity parameter; and

selecting the set of weights associated with [the] <u>a</u> value of the complexity parameter that best satisfies a cross validation criteria;

whereby the selected set of weights used with the modeling function provides the robust model.

2. (Amended) A method of generating a robust model of a system as recited in claim 1 wherein the training error is calculated using a training error criteria that is a function of [the] <u>a</u> difference between training output values associated with training input values determined from the training data set and output values determined from the modeling function and the associated set of weights applied to the training input values.

3. (Amended) A method of generating a robust model of a system as recited in claim 1 wherein the complexity parameter affects [the minimization of] <u>how</u> the training error <u>is minimized</u>.

a

5. (Amended) A method of generating a robust model of a system as recited in claim 4 wherein the complexity of <u>a modeling function having</u> a set of weights is determined by [the squares of the] <u>squared</u> weights <u>of said set</u>.

7. (Amended) A method of generating a robust model of a system as recited in claim 1 wherein the complexity parameter controls an amount of noise that is added to [the] input data of the training set.

12. (Amended) A method of generating a robust model of a system as recited in claim 1 wherein the plurality of values of the complexity parameter are selected to best satisfy the cross validation criteria using [the] <u>a</u> Brent method.

14. (Amended) A method of generating a robust model of a system as recited in claim 1 wherein a threshold is applied to an output of the robust model to classify a set of inputs that generated the output <u>of the robust model</u>.

15. (Amended) A method of generating a robust model of a system as recited in claim 1 wherein the training error <u>for a training data set having input elements and output elements</u> is defined as [the] <u>a</u> sum of [the squares of the] <u>squared</u> differences between <u>said</u> output elements [of the training data] and outputs of the modeling function associated with [each of the] <u>corresponding ones of said</u> input elements [in the training data].

16. (Amended) A method of generating a robust model of a system as recited in claim 1 wherein the training error <u>for a training data set having input elements and output elements</u> is defined as [the] <u>a</u> sum of [the] differences between <u>said</u> output elements [of the training data] and outputs of the modeling function associated with [each of the] <u>corresponding ones of said</u> input elements [in the training data].

17. (Amended) A method of generating a robust model of a system as recited in claim 1 wherein the training error for a <u>training data set having input elements and output elements</u> is defined as [the] <u>a</u> maximum difference between output elements of the training data

a

and outputs of the modeling function associated with [each of the] corresponding ones of said input elements [in the training data].

22. (Amended) A method of generating a robust model of a system as recited in claim 1 wherein the cross validation criteria is minimizing a measure of error between the robust model and the cross validation set.

23. (Amended) A method of generating a robust model of a system comprising:

selecting a modeling function having a set of weights wherein the modeling function has a complexity that is determined by a complexity parameter;

for a each of a plurality of values of the complexity parameter, determining an associated set of weights of the modeling function such that a training error is minimized for a training data set;

determining a cross validation error for a cross validation data set for each set of weights associated with one of the plurality of values of the complexity parameter;

determining an optimal value of the complexity parameter that minimizes the cross validation error; and

determining an output set of weights of the modeling function using the [determined] optimal value of the complexity parameter and an aggregate training data set that includes [the] a training data set and the cross validation data set such that an aggregate training error is minimized for the aggregate training data set; and

whereby the output set of weights used with the modeling function provides the robust model.

24. (Amended) A robust modeling engine comprising:

a memory configured to store a training data set and a cross validation data set;

a processor configured to:

a

select a modeling function having a set of weights wherein the modeling function has a complexity that is determined by a complexity parameter;

for each of a plurality of values of the complexity parameter, determine an associated set of weights of the modeling function such that a training error is minimized for a training data set;

determine an error for a cross validation data set for each set of weights associated with one of the plurality of values of the complexity parameter; and

select the set of weights associated with [the value of] the complexity parameter that best satisfies a cross validation criteria; and

an output configured to output the set of weights associated with the value of the complexity parameter that best satisfies a cross validation criteria.

a